

## REMARKS

Applicant expresses appreciation to the Examiner for consideration of the subject patent application. This amendment is in response to the Office Action mailed June 15, 2005. Claims 1-21 were rejected. The claims have been amended to address the concerns raised by the Examiner.

Claims 1-21 were originally presented. Claims 1-21 remain in the application. No claims have been canceled. Claims 1, 9, 17, and 18 have been amended. No claims have been added.

The antecedent basis for the amendment in claims 1, 9, 17, and 18 can be found in the specification in the paragraph on page 13 starting at line 11.

### **Claim Objections**

Claims 1, 9, and 18 were objected to for antecedent basis errors. The Office Action states that there is a missing “the” before “remaining object files”. However, this is the first time that “remaining object files” appears in these independent claims. Therefore, Applicant respectfully submits that it is not proper for “the” to be used the first time the term “remaining object files” is mentioned.

### **Claim Rejections - 35 U.S.C. § 103**

Claims 1-16 and 18-21 (including independent claims 1, 9, and 18) were rejected under 35 U.S.C. § 103 as being unpatentable over Ching (USPN 5,560,620), hereafter “Ching” in view of Leblang et al. (USPN 5,649,200), hereafter “Leblang”.

The present invention, as claimed in amended independent claim 1, is a method for verifying a version of each of a plurality of object files in a computer program at runtime. An object file is machine code generated by a compiler after it processes a source code file. When different versions of software developer kits are used to generate object files, subtle program errors can result when a computer program uses object files having differing versions. Thus, the present invention enables a version number of the software developer kit to be placed within each object file and enables each object file in a computer program to be checked when the computer program is executed. The files are checked to determine that the object files were generated by

the same software developer kit as the other object files that comprise the computer program. If comparisons show two or more versions, an error message can be generated alerting to the problem or execution of the program may be stopped.

Even if the structures of the Ching and Leblang references are combined as proposed by the Examiner, the result would necessarily constitute a structure different from that of the applicant and one that would not accomplish the result of the claimed invention. The Examiner states that the common endeavor of Ching and Leblang is “to have persisted versions of files that can be reused for subsequent application so that systematic rebuild of files without comparison would not be obviated, thus making the reuse effort less extensive.”

However, the result of the present invention, as claimed in claim 1, is not to have persistent versions of files that can be reused. Rather, as explained in the application, one embodiment of the present invention is to reduce or eliminate bugs in programs caused when different versions of software developer kits are used to compile programs. The present invention does not check to see if a version of an object file has been previously compiled, as taught in Leblang. The previous compilation of an object file is irrelevant to the present invention. Rather, claim 1 reads, in part:

**comparing the version of the selected object file with a version of each of remaining object files of the plurality of object files when the computer program is executed; and**

**generating an alert in response to the version of the selected object file being different than one or more versions of the remaining object files.**

Thus, the object of one embodiment of the present invention is to assure that at computer program execution time the object files in a computer program were all compiled using the same version of a software developer kit or compiler. Neither Ching nor Leblang, nor a combination of the two references teach or suggest this result. Using the teachings of Ching and Leblang, one skilled in the art would not be enabled to determine whether an executable computer program was comprised of object files compiled by differing versions of a software developer kit.

Therefore, Applicant respectfully submits that independent claim 1 is allowable, and urges the Examiner to withdraw the rejection.

The same arguments can be applied to independent claims 9 and 18, which are similar in structure to claim 1.

Rejection of the dependent claims 2-8, 10-16, and 19-21 should be reconsidered and withdrawn for at least the reasons given above with respect to the independent claims. The dependent claims, being narrower in scope, are allowable for at least the reasons for which the independent claims are allowable.

Independent claim 17 was rejected under 35 U.S.C. § 103(a) as being unpatentable over Bowman-Amuah (US Pub. No. 2001/0052108), hereafter “Bowman”, in view of Ching (USPN 5,560,620), hereafter “Ching”.

Bowman discloses an architectural framework for developing a project. (See Bowman Page 1, ¶ 0008 – 0011). Bowman does not disclose a time stamp used to identify a version of a software developer’s kit for each of a plurality of object files in an application. Bowman merely discloses using time stamps at a help desk to assist in logging and processing inquiries to the help desk. (Page 68, ¶ 2165-2172).

Bowman does not teach or suggest a module configured to compare embedded time stamps within the plurality of object files. Finally, Bowman does not disclose a way to generate an error in response to two or more of the object files having different time stamps.

Similarly, Ching does not teach or suggest a means for determining whether an executable application uses object files compiled with different software development kits, as previously discussed.

The combination of structures of the Ching and Bowman references as proposed by the Examiner would result in a structure that is necessarily different from the embodiments of the invention claimed by the Applicant, and one that would not accomplish the result of the claimed invention. Neither Ching nor Bowman, nor their combination teach or suggest a software development kit configured to facilitate a development of an application configured to generate an error when the application is executed in response to two or more of the object files having different time stamps. Using the inventions taught or suggested in Ching and Bowman, it would

not be possible to determine in an executable file whether the program executable included object files that were compiled using different software developer kits. Thus, it would not have been obvious for the Applicant to arrive at the present invention using the teachings or combination of Ching or Bowman.

Therefore, Applicant respectfully submits that independent claim 17 is allowable, and urges the Examiner to withdraw the rejection.

## CONCLUSION

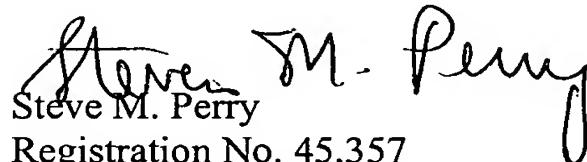
In light of the above, Applicant respectfully submits that pending claims 1-21 are now in condition for allowance. Therefore, Applicant requests that the rejections and objections be withdrawn, and that the claims be allowed and passed to issue. If any impediment to the allowance of these claims remains after entry of this Amendment, the Examiner is strongly encouraged to call Steve Perry at (801) 566-6633 so that such matters may be resolved as expeditiously as possible.

No claims were added. Therefore, no additional fee is due.

The Commissioner is hereby authorized to charge any additional fee or to credit any overpayment in connection with this Amendment to Deposit Account No. 08-2025.

DATED this 13<sup>th</sup> day of September, 2005.

Respectfully submitted,

  
Steve M. Perry  
Registration No. 45,357

THORPE NORTH & WESTERN, LLP  
Customer No. 20,551  
P.O. Box 1219  
Sandy, Utah 84091-1219  
Telephone: (801) 566-6633